DATABASE MANAGING APPARATUS AND DATABASE RECORD RETRIEVING APPARATUS

CROSS REFERENCE TO RELATED APPLICATION

This application is based upon Japanese Patent Application No. Hei. 11-117951 filed on April 26, 1999, the contents of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

1. Field of the Invention:

This invention relates to database managing apparatuses and database record retrieving apparatuses, and particular to a database managing apparatus which makes a database file in which compressed requested records are stored and to a database record retrieving apparatus which extracts a requested record from a memory medium in which a database file is stored.

2. Related Art:

Recently, databases have been used in, for example, a vehicular navigation system as a map database or a telephone number database for setting a destination. The telephone number database includes several attributions of a shop or a company including: a name, a telephone number, an address, and position data (the longitude, the latitude, the altitude). When a driver of a vehicle inputs a search key such as a telephone number of a shop as a destination, the database is searched. When a record corresponding to the search key exists in the database, the navigation system extracts (retrieves) one article data (each attribution) to which the record belongs, and displays information regarding the one article data or uses as the destination.

Here, since the above-described databases including the map database and the telephone number database are generally stored in one common storing medium such as one CD-ROM, an amount of total data of each database is naturally limited. Therefore, the records to be stored are generally stored in the CD-ROM after being compressed (encoded) so as to reduce a size. When the records are read out from the database to be used in a process, the records are restored (decoded) by being decompressed to a former condition.

Here, in a conventional database managing apparatus, the data is compressed by a unit of record in a lump. Therefore, when the database is searched, it needs to decompress by a unit of the record in a lump before searching the database file. In general, a size of a main memory of the CPU for performing the data compression, the data decompression, or the search process, is approximately 10 MB (mega bytes), however, a size of the total records in the database is approximately 100 MB. Therefore, when the CPU performs the search process, the CPU needs to repeat a series of processes including:

(1) loading (reading out) parts of the compressed data from the CD-ROM to the main memory;

(2) decompressing the loaded compressed data to restore it; and

(3) searching the restored data.

Hence, it takes long time to complete searching for one article record.

Furthermore, in a relation model database, which has plural attributions for one article, a projection may be performed. The projection retrieves only particular record having desired attribution

(field). In such a case, it also needs to restore entire records to obtain only particular record belonging to one attribution.

SUMMARY OF THE INVENTION

This invention has been conceived in view of the background thus far described and its first object is to immediately obtain desired data from a database.

According to the present invention, attribution record group forming means classifies data, which is requested to be stored in a database, according to attributions defined in the database, and makes plural attribution record groups corresponding to each of the attributions. Data compressing means compresses the attribution record groups in a unit of each of the attribution record groups. File forming means combines each of the attribution record groups, which is compressed by the data compressing means, and for forming a database file. As a result, the database managing apparatus can reduce needless step for decompressing record data belonging to other attributions than the attribution to be searched for. Furthermore, it can retrieve the requested record in a short time.

According to another aspect of the present invention, a database record retrieving apparatus retrieves a target record to be searched from a database file, which is made up of plural attribution record groups (A-E) each of which is compressed in a unit of each of the attribution record groups. Data decompressing means decompresses a particular attribution record group, which is to be searched, when a search request for searching the database file is received. Searching means searches for a target record containing a search key in the particular attribution

record group. Here, the data decompressing means further decompresses the other attribution record groups, which are different from the particular attribution record group, when the searching means finds the target record. As a result, the database managing apparatus can reduce needless step for decompressing record data belonging to other attributions than the attribution to be searched for. Furthermore, it can retrieve the requested record in a short time.

BRIEF DESCRIPTION OF THE DRAWINGS

These and another objects, features and characteristics of the present invention will be appreciated from a study of the following detailed description, the appended claims, and drawings, all of which form parts of this application. In the drawings, same portions or corresponding to portions are put the same numerals each other to eliminate redundant explanation. In the drawings:

FIG. 1 is a schematic block diagram illustrating a database managing apparatus of a first embodiment according to the present invention;

FIG. 2 is a block diagram illustrating each function realized by software;

FIG. 3 is a flowchart illustrating a data compression process and a database file forming process;

FIG. 4A is an example of a telephone number database before being compressed;

FIG. 4B is an example of the telephone number database after being compressed;

FIG. 5 is a flowchart illustrating a search process;

FIG. 6 is a flowchart illustrating a data decompression process;

FIG. 7 is a flowchart illustrating a data decompression process when a CPU performs a field retrieve;

FIG. 8 is a schematic block diagram of a vehicular navigation system of a second embodiment according to the present invention;

FIG. 9 is a flowchart illustrating a data compression process and a database file forming process of a third embodiment according to the present invention;

FIG. 10 is a flowchart illustrating a data compression process and a database file forming process of a fourth embodiment according to the present invention;

FIG. 11 is a flowchart illustrating a search process of the fourth embodiment; and

FIG. 12 is a flowchart illustrating a data compression process and a database file forming process of a fifth embodiment according to the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

(First Embodiment)

Hereinafter, a first embodiment in which the present invention is applied to a telephone number database for a vehicular navigation system will be explained with reference to FIGS. 1 to 7.

As shown in FIG. 1, which shows an electrical component of a database managing apparatus, the database managing apparatus is provided with a central processing unit (CPU) 1; an input/output (I/O) device 2 connected to the CPU 1 and is made up of a keyboard, an input/output data file or the like; a main memory 3 connected to the CPU 1 and is

made up of a RAM or the like; and an external memory 4 such as a hard disk unit connected to the CPU 1.

As shown in FIG. 2, which shows a block diagram illustrating each function realized by CPU 1 by using software, an application program 5 is a data register program for registering a database record to a database. A database management system (DBMS) 6 belongs to the database managing apparatus, and has functions for classifying the records based on attributions, for compressing the data, and for decompressing the data.

A data controller (data compressing means, data decompressing means, searching means) 7 has a function for operating the input/output device 2, and controls a data compression portion (data compressing means) 8 (1, 2, ..., n) for performing data compression processes (compression methods) and a data decompression portion (data decompressing means) 9 (1, 2, ..., n) for performing data decompression processes (decompression methods).

A database file 10 and a database definition file 11 are stored in the external memory 4. The database definition file 11 includes several definition data regarding the database file 10, that is, data indicating types of the attributions, or data regarding data compression of each data record having individual attribution. After the data controller 7 reads out definition data in the database definition file 11 via a definition file input/output portion 12, the data controller 7 controls the data compression portion 8 and the data decompression portion 9 to perform the data compression process and the data decompression process in accordance with the read definition data.

The data controller 7 accesses the database file 10 via a record input/output portion 13, so as to store compressed data or to retrieve

the stored compressed data. These application program 5, the DBMS 6, the controller 7 etc are stored in the external memory 4. The CPU 1 executes the one of these functions 5-7 by loading onto the main memory 3, if needed.

Next, operations of this embodiment will be explained with reference to FIGS. 3 to 7. FIG. 3 shows a flowchart illustrating the data compression process and the forming process of the database file 10, each of which is executed by the CPU 1, when a record to be stored is provided to the database file 10 via the input/output device 2. FIGS. 4A and 4B show data tables to be stored in the data file 10 as the record. Specifically, FIG. 4A shows the data table before being encoded (compressed), and FIG. 4B shows the data table after being decoded (decompressed). Here, FIG. 4B shows data image of the compressed or encoded condition, but does not show a compression result using a special compression method.

In FIG. 3, the controller 7 firstly obtains, for example, one article record from records, which are read by the application program 5 from the input/output device 2 (step A1). The obtained record data is loaded onto a working area provided on the main memory 3. Next, the controller 7 reads out definition data in the database definition file 11 via the definition file input/output portion 12 (step A2). Then, the controller 7 classifies the records obtained at step A1 according to the attributions based on the definition data, and forms attribution record groups (step A3)(attribution record group forming means, attribution record group forming step).

Here, as show in FIGS. 4A and 4B, in this embodiment, the attributions of the database includes the following attributions A-E

(in the case where the number of the attributions n=5).

Attribution A: Article number

Attribution B: Position data

Attribution C: Name

Attribution D: Telephone number

Attribution E: Address

Therefore, in this case, five attribution record groups corresponding to the attributions A-E (see FIGS. 4A and 4B) are formed.

Next, the controller 7 performs a loop of steps A4-A7, so that the data compression portion 8 (1, 2, ..., n) performs each data compression process according to each attribution record group (steps A5, A6-1, A6-2, ..., A6-n)(data compression step).

The data compression process is performed by using, for example, a LZ (Lempel-Ziv) slide dictionary method, which is suitable for compressing text data. A summary of the data compression process is as follows.

(1) The controller 7 detects whether an address of a data string, which is the same as initial two characters of a data string to be compressed, is registered in a data table of a "dictionary", by using a searching method such as a binary search, a B-Tree search, or a hash search. When such the address is not registered, the controller 7 registers own initial address.

(2) When at least one data string is registered in the dictionary, the controller 7 compares the data string to be compressed with each of the registered data, so as to detect to what order of the character from an initial character does the data string to be compressed match that of each registered data.

(3) When three or more characters are matched as a result of process (2), the controller 7 performs the data compression. In detail, the controller 7 replaces the initial portion of the data string to be compressed with a matching data, which includes an initial address and the number of matched characters (data) of particular data string with which the number of matching is the largest. Furthermore, the controller 7 registers the initial address of this data string to be compressed into the dictionary, and sets a bit, which is provided every plural character and indicates which character in order is not character data but is the matching data.

On the contrary, when the maximum number of the matching is to or less than two, the controller 7 does not perform the compression and directly output the data to be compressed.

The controller completes the data compression by repeating the above-described processes (1)-(3).

Here, the database definition file 11 also includes data that to which attribution record in the attributions A-E does the record to be used as the search key belongs. In this case, for example, when the record of the attribution A is used as the search key, the record group of the attribution A correspond to the record group to be searched. Therefore, the step A6-1 is a step for compressing the record group of the attribution A, however, the controller 7 does not perform the data compression process (the data compression portion 8(1) is not executed) to immediately perform the search process. In FIG. 3, the data compression portion 8(1) is shown by a dotted line.

Regarding the other record group belonging to the attributions B-E, the controller 7 respectively performs the data compression process

in steps A6-2, A6-3, …, A6-n. In detail, as shown in FIG. 4B, each of the record belonging to the attributions B-E is encoded, so that the data size is reduced. In this figure, each of the attributions of the records after compressing are shown as Ac-Dc (however, Ac is substantially the same as A).

When the data compression process regarding the record group of the attribution E ends at step A6-5, the controller 7 escapes the loop of steps A4-A7, and combines the data compressed according to each attribution record group. After that, the controller 7 writes the combined compressed data into the database file 10 via the definition file input/output portion 13 (step A8) (file forming means, file forming step), and terminates the process.

Here, the database file 10 may be directly formed in the external memory at this stage, or may be formed by the following step. That is, the controller 7 makes the database file 10 on the main memory 3 every time the controller 7 processes one article record, and transfers the database file 10 from the main memory 3 to the external memory 4 after plural processes regarding plural articles of the record have completed.

When the step A1 is performed only for a part of data that can be loaded from the input/output device 2 to the main memory 3 at one time, and when the data compression process is performed for not all the requested records, the controller 7 returns to step A1 to repeat the same steps, so as to form the telephone number database in the database file 10.

FIG. 5 shows a flowchart illustrating the search process executed by CPU 1, when there is a search request for a particular record in the database file 10. For example, it assumes that there is a search

request for a particular article whose article number is "1000" in FIGS. 4A and 4B. Here, a search key "1000" is inputted via the input/output device 2. In this case, the controller 7 retrieves parts of the record group belonging to the attribution Ac as the record to be searched from the database file 10 to the main memory 3 (step B1). The binary search, the B-Tree search, or the hash search may be used for searching the database file 10. Since the attribution A is not encoded (compressed), the controller 7 immediately performs the search process after receiving the records from the database file 10.

When there is no record corresponding to the search key "1000", the controller determines as "NO" at step B2, and moves to step B3. When the search for all of the records is not ended at that time, the controller 7 determines as "NO", and moves to step B1 to obtain the other record groups in the attribution A.

On the contrary, when the search for all of the records is ended, the controller determines that there is no record corresponding to the search key in the database file 10, and notices for indicating no corresponding record (step B4). The notice may be performed by displaying information that there is no corresponding record, or by using voice sound. Here, the steps B1-B3 correspond to searching means and searching step.

On the contrary, when there is the record corresponding to the search key "1000", the controller 7 determines as "YES", and performs the data decompression process (step B5).

FIG. 6 shows a flowchart illustrating the data decompression process (data decompressing means, data decompressing step). In this figure, the data controller 7 reads out parts of the record group of

the attributions Bc-Ec, which are other parts than the attribution Ac, from the database file 10 to the main memory 3 (step C1).

Here, since an amount of data is large, the data compression process does not compress entire parts of each attribution record group at one time, but compresses each of the attribution record groups by a particular block, which is divided into an adequate amount of data. Therefore, when a certain article record is searched for, the controller 7 searches only particular block, to which the certain article belongs, in the database file 10.

Next, the controller 7 reads out the definition data in the database definition file 11, alike the step A2 (step C2). The controller 7 classifies the records obtained at step C1 according to the attributions based on the definition data (step C3). Then, while the controller 7 repeats a loop of steps C4-C7, the controller 7 performs the data decompression process every attribution record group by using the data decompression portion 9 (2, ..., n) (steps C5, C6-2, ..., C6-n).

After the data of the record groups belonging to the attributions B-E are restored (decompressed), the controller 7 outputs the article records (attributions A-E) to the input/output device 2, and terminates the process.

FIG. 7 shows a flowchart illustrating the data decompression process (data decompressing means, data decompressing step), when only data belonging to a particular attribution is selectively searched (field search). For example, in a case of the vehicular navigation system, when the destination is set, the names of the shops or companies may be listed on the display and may be selected by moving a cursor or the like. The field search is performed in such a case.

In this figure, when particular attribution data (e.g., attribution C), which is requested for being searched for, is inputted via the input/output device 2, the controller 7 reads out the definition data in the database definition file 11 (step D1). The controller 7 reads out only record group belonging to the attribution Cc, which corresponds to the attribution C and is encoded (compressed), from the database file 10 (step D2). After that, the controller 7 restores the record data regarding the record group belonging to the attribution Cc to the former condition, by using the data decompression portion 9 (3). Then, the controller 7 outputs the restored record data to the input/output device 2.

As describe the above, according to this embodiment, the controller 7 does not compress the data regarding the record group belonging to the attribution A, which is the record group to be searched, in all of the attributions A-E, but compresses only data regarding the record groups belonging to the attributions B-E, which are the record groups other than the record group to be searched.

Therefore, when there is a search request for a particular record, the controller 7 can immediately start the search process without waiting the data decompression process for the record group belonging to the attribution A, and can perform complete the search process in a short time. Furthermore, since the record groups belonging to the attributions B-E are decoded (decompressed) only when there is a corresponding record as a result of the search process, it can reduce needless time for performing the data decompression process and can retrieve the requested record. That is, a process time of this embodiment becomes shorter than a conventional system, in which all of the data

are decompressed before the search process. In addition, it can prevent the size of the database from increasing by performing the search process in a short time.

Furthermore, the controller 7 reads out and searches only the record group belonging to the attribution A from the database file 10, and reads out and decompresses the other record groups belonging to the attributions B-E only when there is the corresponding record in the database file 10 as a result of the search. Therefore, it can reduce the number of access (loads of the data from the external memory to the main memory) to the database file 10 during the search process and during obtaining the all requested records, and can reduce the process time.

Furthermore, according to this embodiment, regarding the records belonging to the attributions B-E, the controller 7 respectively compresses data regarding each attribution record group. When the controller 7 selectively searches only the record group belonging to the attribution C, the controller 7 decompresses only the record group belonging to the attribution C. Hence, it does not need to decompress needless record data belonging to other attributions than the attribution to be searched for. As a result, it can retrieve the requested record in a short time.

(Second embodiment)

FIG. 8 shows a schematic block diagram illustrating an electrical component of a vehicular navigation system (database record retrieving apparatus) of a second embodiment according to the present invention. The vehicular navigation system is provided with a controller (searching means, data decompressing means) 14 made up of a microcomputer or the like, a GPS receiver 15, an azimuth sensor 16, and a wheel speed

sensor 17. The GPS receiver 15, the azimuth sensor 16, and the wheel speed sensor 17 are for receiving GPS signals from the GPS satellite and for calculating a present portion of the vehicle. The vehicular navigation system further includes an operation portion 18, which is provided with operation keys with which a user operates to input data, a display unit 19 such as a display, and a voice generation unit 20 such as a speaker.

A database of map data for being displayed on the display unit 19 and the database file 10 of the telephone number database shown in FIG. 4B in the first embodiment. In detail, the database file 10, which is formed by the database managing apparatus in the first embodiment, is written in a CD-ROM 21 by converting a data format to a CD-ROM format. The data in the CD-ROM 21 can be read out to the controller 14 by using a driver 22.

According to the vehicular navigation system of this embodiment, the controller 14 reads out the data stored in the CD-ROM 21, when it is needed, and the controller 14 perform route guidance by displaying the data on the display unit 19 or by using voice sound. The present invention is applied to one function for reading out and decompressing the telephone number database, which is stored with being compressed.

In detail, the program of the data decompression process as shown FIGS. 6 and 7 in the first embodiment is built in a control program of the controller 14. The controller 14 reads out and decompresses the data from the CD-ROM 21 in accordance with an operation of the user via the operation portion 18. As a result, the data decompression process of this embodiment can be performed by the same way as that of the first embodiment.

As described above, according to this embodiment, the controller 14 of the vehicular navigation system reads out and decompresses the data from the CD-ROM 21, in which the database file 10 formed by the database managing apparatus of the first embodiment is stored. Therefore, it can retrieve the requested record in a short time.

(Third embodiment)

FIG. 9 shows a flowchart illustrating a data compression process and a database file forming process of a third embodiment. Differences between this embodiment and the first embodiment will be mainly explained.

In detail, the step A3 in the first embodiment is replaced with step A3a. At step A3a, the controller 7 classifies the record, which is requested for being stored, into the record group to be searched and the other record groups.

The step A5 is replaced with step A9. At step A9, the controller 7 judges whether the attribution record group is the record group to be searched. When the attribution record group is the record group to be searched, the controller 7 determined as "YES", and moves to step A7 without performing the data compression process, alike the first embodiment. On the contrary, when attribution record group is not the record group to be searched, the controller 7 determined as "NO", and performs the data compression process by using the same way each other among the attributions (step 10). Here, the steps A9 and A10 correspond to the data compressing means and the data compressing step.

As described above, when there is a search request for a particular record in the database file 10, the controller 7 performs the search process, and reads out and decompresses the data belonging to the attributions Bc-Ec of the particular record if there is a

corresponding record. According to the this embodiment, even when there are many such processes, the controller 7 can perform the data decompression process regarding the attributions Bc-Ec in a lump, the controller 7 can retrieve the data in a short time.

(Fourth embodiment)

FIG. 10 is a flowchart illustrating a data compression process and a database file forming process of a fourth embodiment. FIG. 11 is a flowchart illustrating a search process. Differences between this embodiment and the first embodiment will be mainly explained.

In detail, in the first embodiment, the record belonging to the attribution A as the record group to be searched is not compressed. However, in this embodiment, as shown in FIG. 10, the controller 7 also performs the data compression process to the record group belonging to the attribution A at step A6-1 by using the data compression portion 6(1).

In the search process, as shown in FIG. 11, after the controller 7 obtains the data of the record group belonging to the attribution Ac at step B1, the controller 7 performs the data decompression process with respect to the obtained data (step B6). When there is a corresponding record, the controller 7 determined as "YES", and performs the data compression process with respect to the other record groups belonging to the attributions Bc-Ec (step B5a).

As describe the above, since the controller 7 also compresses the record group belonging to the attribution A as the record group to be searched, it can reduce total process time compared to the conventional method in which the search process is performed after all of the records are decompressed. Furthermore, since the record group belonging to the

attribution A is also compressed, the size of the database file 10 can be reduced.

(Fifth embodiment)

FIG. 12 shows a flowchart illustrating a data compression process and a database file forming process of a fifth embodiment. Differences between this embodiment and the third embodiment will be mainly explained.

In the third embodiment, when the controller 7 determines as "YES" at step A9, the controller 7 does not perform the data compression process; and when the controller 7 determines as "NO" at step A9, the controller 7 performs the data compression process.

On the contrary, in a fifth embodiment, when the controller 7 determines as "YES" at step A9, the controller 7 compresses the record group belonging to the attribution A with a high-speed decompressable format, which is a relatively low compression rate format and can be decompressed in a short time (step A11). When the controller 7 determines as "NO" at step A9, the controller 7 compresses the record group belonging to the attribution A with a relatively high decompression rate format (step A12). Here, the steps A9-A11 correspond to the data compressing means and the data compressing step.

As described above, in the data compression process, the "dictionary" is formed for decompressing and restoring the compressed data portion. When a size of the dictionary is set to large, the compression rate can be increased, because a possibility for matching further longer character strings may be increased in the data string to be compressed. On the contrary, when the size of the dictionary is set to small, the compression rate is lowered.

Here, when the compression rate of the data is increased, it takes relatively long time to complete the data decompression process, because a time needed for searching the dictionary during the data decompression process is prolonged. On the contrary, when the compression rate is lowered, the data decompression process can become shorter.

Therefore, the controller 7 restricts the size of the dictionary to small at step A11, so that the compression rate regarding the record group to be searched becomes relatively small, and that the controller can immediately start the search process by reducing the time needed for the data decompression process. On the contrary, the controller 7 sets the size of the dictionary to large at step A12, so as to reduce the size of the other record groups belonging to the other attributions, which are needed only there is a corresponding record in the record to be searched.

As described above, according to this embodiment, since the data compression rate of the record group to be searched is set to small, and that of the other record groups belonging to other attributions is set to high, the controller 7 can perform the search process at high-speed, and can prevent the total size of the compressed data from increasing.

The present invention is not limited to the above-described embodiments, but can be modified and expanded to the following modifications.

The data decompression portion 9 may be omitted from the database managing apparatus of the first embodiment, so that the database managing apparatus mainly includes a function that the database file 10 is formed by performing the data compression process.

When there is no database definition file 11, the controller

7 may classify the attributions of the records to be stored by using a predetermined parameter, and so does the data compression method of each record group belonging to each attribution.

When the controller 7 performs the search process, the controller 7 may load (read out) the data of the other record groups, the other than the record group to be searched, onto the main memory 3.

The database file formed by the data compression method as described in the third to fifth embodiments may be stored in a storing medium such as a CD-ROM, and may be applied to a database record retrieving apparatus such as the vehicular navigation system as shown in the second embodiment, which is provided with the data decompression method corresponding to each of the data compression method.

In the fifth embodiment, the record groups belonging to the other attributions other than the record group to be searched may be compressed regarding each attribution record group, alike the first embodiment.

When there is adequate encode (compression) method for an attribution record group from a viewpoint of a characteristic of each attribution record group, the encode methods may be different from each other.

For example, in the first embodiment, a judgment step for asking the number of the articles is inserted between the steps A3 and A4. The controller 7 repeats the steps A3-A4 until the number of the obtained record articles reaches the number of the articles. When it reaches, the controller 7 determined as "YES" at this inserted step, and performs the data compression process as shown in A4-A7.

When the data string to be compressed has already existed as a file, and when the controller 7 can be obtain plural of article records at the step A1 by reading out the data file regarding each block, the controller 7 may perform the steps A2, A3 and A8 with each block in a lump. Furthermore, when the size of the data string to be compressed is relatively small enough to be read out at one time, each of the steps A2, A3 and A8 may be formed in a lump.

For example, when a particular character string, which is frequently used, such as "corporation" or "co., Ltd" may be expressed by, for example, one byte code, so that a so-called "encode dictionary" is formed by storing the database file 10 in a predetermined region. As a result, the size of the data can be further reduced.

The external memory 4 may be separately formed from the CPU 1, when the external memory 4 is connected to the CPU 1 via a network.

The storing medium is not limited to the CD-ROM 21, but may use a DVD-ROM, an MO disk, a hard disk unit, or a memory card.

The database managing apparatus or the record retrieving apparatus of the present invention is exemplified to the telephone number in the above-described embodiments. However, the present invention may be applied to other apparatus, which searches or retrieves particular data corresponding to particular elements or attributions, such as a bank account managing system, a client database managing system, or sales database managing system.